# A Cost-Effective LabVIEW-Arduino Framework for Control System Computation and PID Simulation in Engineering Education

Rommel G. Gadia[1,2,*] and Robert V. Marcos[2]

[1]Office of University Research Development, Tarlac State University, Tarlac City, Philippines
[2]Department of Electrical and Electronics Engineering, College of Engineering, Tarlac State University, Tarlac City, Philippines
Email: rggadia@tsu.edu.ph (R.G.G.); rvmarcos@tsu.edu.ph (R.V.M.)
*Corresponding author

Manuscript received February 23, 2025; revised March 24, 2025; accepted May 7, 2025; published August 7, 2025

***Abstract*—This study presents a cost-effective LabVIEW-Arduino framework for numerical computation in control systems and Proportional-Integral-Derivative (PID) controller simulation, designed to enhance engineering education. The proposed platform addresses limitations in traditional simulation tools, offering an affordable and accessible alternative for institutions with budget constraints. The research employed a Developmental Research approach using the Analysis-Design-Development-Implementation-Evaluation (ADDIE) model. The system supports key control computations, including transfer function to state-space conversion, zero-pole analysis, transient response identification, and PID control implementation. To evaluate usability, Electronics Engineering students enrolled in a Feedback and Control Systems course used the platform, with assessments conducted using the Post-Study System Usability Questionnaire (PSSUQ). The platform demonstrated high usability and educational value, achieving strong ratings across PSSUQ categories: System Usefulness (2.30), Information Quality (2.45), Interface Quality (2.15), and Overall Satisfaction (2.40) on a scale where lower scores indicate higher satisfaction. The interactive simulations and real-time data visualization enhanced students' understanding of control system concepts, bridging the gap between theory and practice. The LabVIEW-Arduino framework offers a scalable and practical solution for control system education, aligning with Commission on Higher Education Memorandum Order (CHED CMO) 101, series of 2017 curriculum standards. Its low cost and ease of use make it particularly valuable for resource-constrained settings. Future work will explore advanced control algorithms and remote learning capabilities to support modern engineering education further.**

***Keywords*—control system education, LabVIEW-Arduino, Proportional, Integral, and Derivative (PID) simulation, engineering education, simulation-based learning, affordable learning technologies**

## I. Introduction

Engineering, particularly control systems engineering, plays a vital role in modern life by enabling the understanding and manipulation of natural forces and materials for diverse applications. Control systems are among the core principles of engineering technology, ranging from manufacturing to robotics, energy systems, and aerospace applications. Control systems, if taught well, provide engineers with theoretical knowledge and practical skills to design, analyze, and implement such systems proficiently [1]. Control systems theory, encompassing topics such as feedback control, stability analysis, system modeling, and controller design, forms a critical foundation for preventing inefficiencies and failures in engineered systems. Furthermore, integrating multidisciplinary knowledge from mechanical, electrical, and chemical engineering facilitates the development of more robust solutions [2].

On the other hand, simulation-based learning has gained currency as an essential step in control system education to fill the gap between theory and application. It enables students to manipulate system parameters and control strategies in the virtual world, fostering their assisted intuition and problem-solving ability [1]. That is, high-fidelity simulations replicate target complex systems precisely so that students can explore the vast array of scenarios, including conditions that would be practically impossible to recreate in real life. Thus, bringing computational tools into the engineering education system is essential for producing well-trained professionals tackling real-life challenges [3]. Even with simulation-based learning's clear advantages, the specific limitations beset the full effectiveness of simulation methods. One crucial problem is the "reality gap," where simulation results may not fully capture real-world performance due to unmodeled dynamics, sensor noise, or other disturbances in the external environment [3]. Such discrepancies can cause failures for control strategies optimized under simulations when tested in the real world. Besides that, gaining access to good-quality simulation platforms is still quite challenging in resource-constrained institutions. Some commercially available solutions, such as MATLAB/Simulink, are prohibitively costly in licensing, so much so that they are not widely adopted in academia [4].

As technology continues to develop, platforms such as LabVIEW Interface for Arduino (LIFA) have emerged. These platforms facilitate the integration of Arduino with LabVIEW software. Arduino, an open-source platform, plays a vital role in engineering education due to its versatility, affordability, and ease of use. It is also widely used for fast prototyping and other educational projects. On the other hand, the software LabVIEW, which stands for Laboratory Virtual Instrumentation Engineering Workbench, enables the user to execute measurable designs and control and test applications. This software is a system design platform for data gathering, controlling various instruments, and even automation [5]. Its flexibility allows for its application in diverse fields, including electromagnetics applications [6–9], agriculture [9–12], and low-cost learning support material [13–25]. In addition, many hardware-software combinations are available for control systems education, but integrating cost-effective platforms has not undergone much exploration. LabVIEW and Arduino are capable of real-time data acquisition and visualization to make even complex engineering concepts much more digestible [26]. However, little research has contributed effectively to using Arduino and LabVIEW in control systems computation and Proportional-Integral-Derivative (PID) simulation. A Proportional-Integral-Derivative controller (PID controller) is a feedback control mechanism widely used in engineering to maintain a desired system output. It adjusts the control input based on the current error (Proportional), the accumulation of past errors (Integral), and the prediction of future errors (Derivative), ensuring accurate, stable, and responsive system performance. Hence, the possibilities of providing a relatively cheap, interactive, scalable learning tool for engineering students remain unexplored within such a framework.

The study tests the feasibility of a cost-effective LabVIEW-Arduino framework for control system numerical computation and PID controller simulation, addressing the gaps. This study also aligns with the expected learning outcomes defined in CHED CMO 101, series of 2017, which outlines the core competencies for feedback and control systems courses [27]. It performs fundamental computations required for various control system analyses and design tasks, including the conversion of a transfer function—a mathematical representation that defines the input-output relationship of a linear time-invariant system in the Laplace domain—into a state-space model, which expresses the system dynamics using a set of first-order differential equations suitable for modern control methods. Additionally, it facilitates zero-pole analysis, which involves identifying the system's zeros and poles to assess stability and dynamic behavior. It also enables transient response analysis, which examines how the system reacts to changes in input over time, particularly before reaching a steady state. These computations support PID control implementation in typical control applications, such as regulating the speed and position of a DC motor.

Furthermore, integrating interactive simulations with physical hardware aligns with established pedagogical principles emphasizing active learning and experiential engagement. Such approaches empower students to move beyond passive observation, enabling them to actively construct knowledge by experimenting with system parameters, observing real-time responses, and directly connecting theoretical concepts to tangible outcomes. This study proposes a framework to leverage these principles within a cost-effective structure.

In this light, this study aims to design and develop a LabVIEW-Arduino-integrated platform for control system numerical computation and controller simulation. More specifically, (a) To develop a LabVIEW-Arduino framework that supports key control system computations, including transfer function to state-space conversion, zero-pole analysis, transient response identification, and PID control implementation; and (b) To assess the usability of the platform using the Post-Study System Usability Questionnaire (PSSUQ).

## II. Review of Related Literature

### *A. LabVIEW-Arduino for Control System Education*

LabVIEW, or Laboratory Virtual Instrument Engineering Workbench, is a graphical programming environment made to aid scientists and engineers in automating test and measurement applications. Unlike traditional text-based programming languages, LabVIEW fosters the creation of block diagrams, whereby interconnected nodes represent the functions, and wires dictate the data flow [28]. LabVIEW simplifies programming for beginners with its graphical features. Its extensive libraries and virtual instruments enable quick data acquisition, instrument control, and analysis development. The dataflow programming model promotes parallelism based on input availability, while its hierarchical structure enhances modularity and code reuse.

Integrating Arduino with LabVIEW offers significant benefits for control systems education and prototyping. Arduino is an open-source hardware platform that provides inexpensive but versatile microcontrollers with an easy-to-learn programming environment [29]. In combination, LabVIEW's graphical programming and Arduino's hardware interface make designing interactive control systems and data-acquisition applications possible. Communication between LabVIEW and Arduino is easier using tools such as the LabVIEW Interface for Arduino, which allows LabVIEW to send commands, process sensor data, and control actuators.

Several studies have explored the integration of LabVIEW and Arduino, often comparing this combination with other educational platforms. Uyanik and Catalbas [4] presented a low-cost laboratory setup for teaching control theory through Arduino, which is also eminent in MATLAB/Simulink [30]. Their work emphasized extensive hands-on learning, utilizing hardware installations integrated with high-level design tools—a common characteristic in the reviewed literature on LabVIEW and Arduino. Nicols Montes *et al*. [31] encouraged using mobile robot platforms like LEGO EV3 with MATLAB/Simulink for education in robotics. The study by Mustafa Saad *et al*. demonstrates the platform's applicability for educating students in control systems through real-time DC motor position control by LabVIEW employing Arduino [32]. The PID control has closed-loop control, giving the students an experimental understanding of basic control. Another example is reported by Bhaskar Dudem *et al*. [33] who developed a project that used LabVIEW and Arduino in a triboelectric nanogenerator (TENG) project with Morse code communication. It has many benefits while teaching about renewable energy, signal processing, and communication systems. Most importantly, this hybrid platform is cost-effective for using LabVIEW and Arduino in education. Papers were reviewed to highlight the points where cost-effective hardware and software components were emphasized as low-cost integrated systems.

### *B. Advantages of LabVIEW-Arduino*

Using LabVIEW with Arduino in technical projects and educational settings offers several advantages, enhancing practical applications and learning experiences. In technical projects, this combination provides cost-effectiveness and flexibility, as Arduino is a low-cost, open-source platform that, when integrated with LabVIEW, enables the creation of affordable and adaptable data acquisition systems suitable for applications such as control, data collection, and rapid prototyping [34, 35]. Additionally, LabVIEW's user-friendly interface design complements Arduino's ease of programming, allowing for seamless integration that facilitates the development of robust systems for tasks such as temperature control and environmental monitoring [36, 37]. This combination also supports rapid prototyping, enabling quick development and testing of prototypes, which is particularly beneficial for experimental setups and iterative design processes [34, 38].

In educational settings, the integration of LabVIEW and Arduino enhances learning outcomes by helping students grasp data acquisition, control systems, and virtual instrumentation concepts through a hands-on approach, improving their comprehension and retention of complex engineering principles [37, 39]. Furthermore, it fosters practical application and engagement by allowing students to work on real-world projects, such as building and controlling robots or remote monitoring systems, strengthening their problem-solving skills and creativity [40, 41]. Additionally, using LabVIEW's remote panel technology in combination with Arduino supports remote learning, enabling students to conduct experiments and access learning resources from any location—a feature that proves particularly valuable during situations like pandemics [41, 42].

Overall, the integration of LabVIEW with Arduino provides a powerful, cost-effective, and flexible platform for both technical projects and educational purposes. It enhances practical learning experiences, supports remote education, and facilitates the development of innovative solutions across various fields.

### *C. Alternative Low-Cost Control System Platforms*

The increasing complexity of real-world industrial control systems presents significant challenges for control engineering education [43]. Coupled with budget limitations in educational institutions [44, 45], the need for cost-effective solutions for laboratory work has become paramount. Furthermore, events like the COVID-19 pandemic have underscored the importance of remote and accessible laboratory experiences for students [46].

Several research efforts have focused on developing low-cost platforms and frameworks to address these challenges in control engineering education. These initiatives leverage affordable hardware like Raspberry Pi and Arduino [44, 46–48] alongside free and open-source software such as Scilab/Scicos and Linux [43] to create effective learning environments. The motivations behind these developments range from enhancing students' practical understanding of control theory and their ability to apply it to physical systems [49] to providing accessible and safe experimentation opportunities without the need for expensive commercial equipment.

Table 1 provides a cost analysis of various low-cost laboratory systems and platforms discussed in the provided sources, offering insights into the diverse approaches and the associated financial implications for implementing hands-on and remote-control engineering education.

Table 1. Some Educational control system platforms

| Reference | Description | Cost Analysis |
|---|---|---|
| [43] | Focuses on developing a low-cost embedded controller for complex control applications using the free and open-source software Scilab/Scicos on a Cirrus Logic EP9315 ARM9 systems-on-chip board running Linux. | The primary cost reduction is achieved through free and open-source software like Scilab and Linux. While the hardware platform (Cirrus Logic EP9315 ARM9 board) is used, the paper emphasizes the minimization of software development costs which have become dominant. These excerpts do not detail specific hardware costs for a complete control system. |
| [44] | A cost-effective remote laboratory for process control education built on Raspberry Pi and Arduino, utilizing various open-source software technologies. Provides three thermal plants, one magnetic levitation, and one hydraulic tank system. | The project had a budget of €1000, with approximately €500 spent on the architecture development, excluding IP cameras. Individual component costs are not detailed, but the emphasis is on "very cheap hardware components" |
| [46] | Describes a low-cost remote laboratory for tank level control using Raspberry Pi, acrylic tanks, an ultrasonic sensor, and motor pumps. Students can program digital controllers in Python. | The paper provides a detailed hardware cost breakdown, totaling $165.5 USD, for the main components of a two-tank system. This includes the control plant, Raspberry Pi, camera, motor pumps, sensor, amplifier, power supply, motor driver, and ADC. |
| [47] | Abstracts a paper about a portable low-cost Arduino-based laboratory kit designed for control education. It aims to teach fundamental control concepts. | The abstract indicates a total equipment cost of 20-25 Euros for the kit, making it a very affordable option for providing hands-on experience. |
| [48] | Presents a very low-cost laboratory setup for feedback control systems education using an Arduino microcontroller and Matlab/Simulink interface, focusing on DC motor control. | A single experimental kit costs approximately $97.06 USD, with a detailed breakdown of component costs provided, including Arduino Uno, motor shield, DC motor with encoder, power adapter, and mechanical components. This low cost allows for providing individual setups for students. |
| [49] | Presents a low-cost and high-safety design framework for virtual control systems education using digital twins of physical systems implemented on microcontrollers. Allows students to design and tune controllers for various virtual systems. | The framework emphasizes cost-effectiveness (P1) by enabling the digital generation of different physical systems, thus minimizing the need for expensive physical equipment. No specific cost figures for implementing the framework are provided, as it focuses on a methodology. |
| [50] | Introduces the concept of a low-cost remote laboratory based on National Instruments NI myDAQ for electronic engineering education. It features a breadboard and interfacing capabilities. | Mentions myDAQ as a cost-efficient hardware platform suitable for prototyping and developing remote exercises. However, the excerpts do not provide a specific cost analysis for a complete remote lab setup using myDAQ |

## III. Materials and Methods

In this study, the ADDIE model—comprising Analysis, Design, Development, Implementation, and Evaluation—was the guiding framework for developing the LabVIEW-Arduino platform (Fig. 1). The choice of the ADDIE model was driven by its structured yet flexible approach, enabling systematic planning, development, and evaluation of the instructional tool and its associated learning experiences. The following is a description of each step of the research.


Fig. 1. ADDIE model.

During the Analysis phase, the need for a cost-effective, accessible control system simulation tool was identified. This phase involved understanding the limitations of existing simulation tools, especially in resource-constrained institutions. The topics covered in the proposed system are based on CHED CMO No. 101, series of 2017.

The architecture for the system was designed during the design phase and focused on user-friendly interfaces and practical applications in control system education. Functionality was further elaborated to ensure the platform performs basic control computations such as converting transfer functions, zero-pole analysis, evaluating the transient response, and PID control. Appropriate survey instruments for evaluating the platform's relevance and impact were also identified or designed during this phase.

In the Development phase, construction of the LabVIEW-Arduino platform began, integrating both hardware and software aspects. A printed circuit board (PCB) was designed to accommodate the Arduino Mega and motor drive. At the same time, a GUI for control and real-time data visualization of LabVIEW software was being developed.

The system was implemented and tested in a Feedback and Control Systems course at Tarlac State University in the Implementation phase. Electronics Engineering students interfaced with the platform to perform assigned tasks, such as PID simulation and state-space analysis. Their experience using the system was monitored for functionality assessment and impact on education.

The last phase of the evaluation used the Post-Study System Usability Questionnaire (PSSUQ) for platform usability and user satisfaction assessments. The results led to refinements in the platform to ensure alignment with educational needs and from a practical perspective.

Through the iterative implementation of the ADDIE model, this study systematically developed and evaluated a viable and inexpensive platform for control systems education that adheres to acceptable standards and meets student expectations.

### A. Post-Study System Usability Questionnaire

To assess acceptability, Electronics Engineering students from the Tarlac State University College of Engineering, enrolled in the Feedback and Control Systems course, participated as respondents. These participants interacted with the platform and subsequently completed the Post-Study System Usability Questionnaire (PSSUQ), developed by Lewis (1992) [51]. The metrics are organized into 16 questions that fall under four primary categories: information quality (INFOQUAL), system usefulness (SYSUSE), interface quality (INTERQUAL), and overall satisfaction score (OVERALL). The PSSUQ score ranges from 1, which signifies strong agreement, to 7, which indicates strong disagreement [52]. The questionnaire demonstrated criterion validity, with a moderate correlation coefficient ($r = 0.80$) to other user satisfaction measures and high reliability, as evidenced by a coefficient alpha of 0.96 [53].

Table 2. PSSUQ statements

| No. | Questions |
|---|---|
| 1 | I really like with this system's general ease of use. |
| 2 | This way of working was easy to use. |
| 3 | Through this process, I was able to finish all the tasks and simulations quickly. |
| 4 | I had no issues adopting this system. |
| 5 | This system was simple to learn how to take advantage of. |
| 6 | Considering this setup, I think I could get consumed with soon. |
| 7 | Error messages from the system made it very evident to me how to resolve issues. |
| 8 | Once I erred while using the technique, I was able to get back up swiftly and effortlessly. |
| 9 | This system came with clear documentation, on-screen messages, and online support, among additional features. |
| 10 | Finding the information, I desired was simple. |
| 11 | The completion of the exercises and scenarios was aided by the information. |
| 12 | The information on the platform's panels was clearly organized. |
| 13 | This system had a great user interface. |
| 14 | I particularly enjoy using this system's interface. |
| 15 | This system fulfills all my expectations in terms of features and functionality. |
| 16 | Overall, I am impressed by this system all around. |

The table contains 16 statements that users respond to, each highlighting different aspects of the system's usability, functionality, and overall experience. The first set (1–6) focuses on general ease of use, efficiency, and the user's ability to complete tasks, evaluating the system's intuitiveness. Statements 7–9 assess the clarity of error messages, how easily users can recover from mistakes, and the quality of support documentation and on-screen help. These aspects are crucial for determining how well the system aids users in overcoming challenges. The next set (10–12) examines how accessible and organized information is within the system, including how easily users find what they need and how effectively it helps them complete tasks. The final statements (13–16) evaluate the overall user interface (UI), user satisfaction with features, and general impressions of the system. These responses shed light on the system's design quality and its ability to meet user expectations. Overall, the PSSUQ statements in Table 2 address various usability aspects, providing insights into where the system excels or needs improvement [54].

### B. Data Collection and Analysis

A total of 22 Electronics Engineering students enrolled in the Feedback and Control Systems course at Tarlac State University participated in the usability assessment. Following their hands-on interaction with the LabVIEW-Arduino platform, the students completed the PSSUQ, which was administered as a printed questionnaire. Participants were instructed to reflect on their genuine experiences with the platform to minimize response bias, emphasizing its strengths and improvement areas. All responses were collected anonymously to ensure honest feedback. Participants provided informed consent before participation.

Statistical analysis employed total enumeration since it included the entire target population at the locale; hence, descriptive statistical methods were applied to analyze the data. The mean scores for each PSSUQ category (SYSUSE, INFOQUAL, INTERQUAL, OVERALL) were calculated to identify areas of high satisfaction (scores close to 1) and potential improvement (scores approaching 7). The overall mean score across all categories was also computed to gauge general user satisfaction.

### C. Ethical Considerations

This study prioritized data confidentiality, participant rights, and ethical research standards. Informed consent was explicitly obtained, ensuring participants were fully aware of the study's objectives, their voluntary involvement, and their right to withdraw at any time.

## IV. Results

### A. Proposed System

The proposed system used in this study focused on the feedback and control system course topics implemented through the two-part system, which are the hardware and software platforms. Fig. 2 shows the block diagram and an overview of how components are interconnected regarding power (red lines) and data communication (blue lines) between the Arduino, Motors, Sensors, and computer.

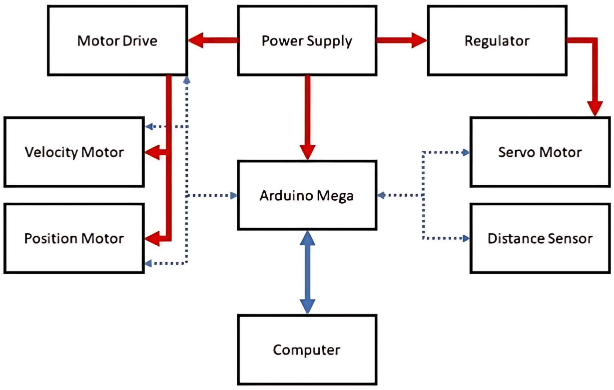


Fig. 2. Block diagram.

The system begins with a Power Supply that provides electrical power to the motor drive and voltage regulator. The Arduino Mega connects the LabVIEW executable on the user's computer to the components, while the distance sensor functions as a feedback loop in the Beam Balancer system. This setup facilitates PID simulations for velocity and position motors, with the computer displaying time-based graphs of PID performance and numerical calculations. A Proportional-Integral-Derivative (PID) controller is a fundamental feedback control mechanism extensively utilized in engineering and automation to maintain a system's output at a desired setpoint. It operates by continuously calculating an error value, representing the difference between the desired setpoint (SP) and the measured process variable (PV). Based on this error, the PID controller computes a corrective output signal to adjust the system, which is a weighted sum of three distinct control terms. The Proportional (P) Term provides a control action directly proportional to the current error, responding to the present magnitude of the deviation and initiating immediate corrective measures; its contribution is calculated by multiplying the current error by a proportional gain constant. The Integral (I) Term considers the accumulation of past errors over time, with its primary function being to eliminate steady-state errors—persistent offsets that might remain with only proportional control—by integrating the error until it is driven to zero; this is achieved by summing the error over a period and multiplying it by an integral gain constant. Finally, the Derivative (D) Term anticipates future errors by reacting to the rate at which the error is changing, providing a damping effect on the system's response that can reduce overshoot and improve stability during transient periods; this component is calculated based on the derivative (or rate of change) of the error signal, multiplied by a derivative gain constant. These three components—Proportional, Integral, and Derivative—work in concert to enable the PID controller to achieve smooth, stable, and accurate regulation of the system. Due to their robustness and effectiveness, PID controllers are pervasively employed in a vast array of applications, including temperature control, motor speed and position regulation, robotics, and numerous other automated processes.

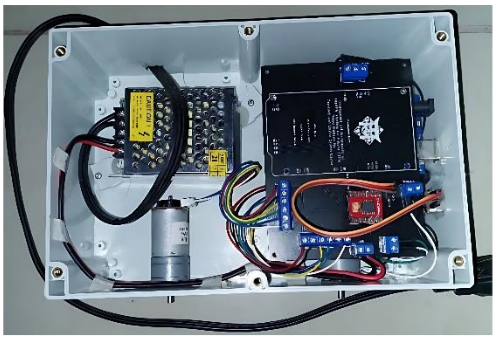

Fig. 3. Glimpse of the project with the designed PCB.

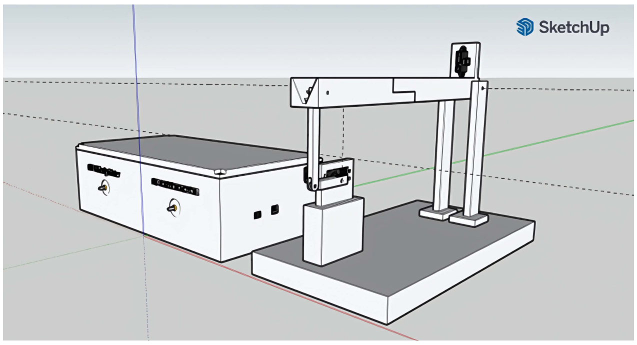


Fig. 4. Isometric perspective of the 3D project render with the beam balancer.

Using KiCAD, the circuit was designed for a Printed Circuit Board (PCB) measuring 96mm x 126mm, which accommodates an Arduino Mega and motor drive. The PCB features header pins and terminal jacks for motors and sensors on the top. The prototype is housed in an IP65 enclosure measuring 240mm × 160mm × 90mm, which contains the PCB, power supply, and motors. Fig. 3 displays the actual PCB and its connections. The top view of the said figure highlights the I/O pin and USB ports, with labels for the beam balancer.

The beam balancer structure (Fig. 4) is 3D printed. The distance sensor and servo motor, which form the core of the beam balancer mechanism, are connected to the input/output pins of the Arduino Mega. The Arduino, in turn, communicates with the LabVIEW software on the computer via a USB connection. Technical specifications for the components are in Table 3.

Table 3. Specification of materials used

| Parts | Specification |
|---|---|
| Arduino Mega 2560 R3 | Based on ATmega 2560 microcontroller with 5V operating voltage and 16MHz clock together with 54 digital I/O pins and 16 analog input pins |
| DC Motor | 12V operating voltage encoder motor |
| Motor Driver Module | Capable of controlling 2 DC motors at 1.2A constant current with a maximum of 15VDC supply voltage at 2.2VDC to 5.5VDC motor supply voltage operation |
| Servo Motor | With 4.8VDC to 6VDC operating voltage with 0.15 to 0.19 seconds per 60 degrees speed corresponding to 94 to 11 kg/cm stall torque |
| Distance Sensor | An IR-based sensor with a 4cm to 30cm distance range at a 60Hz sampling rate operating at 4.5VDC to 5.5VDC |

Moreover, Table 4 presents the bill of materials. One of the proposed control system platform's key strengths is its affordability, making it accessible to academic institutions with limited budgets.

Table 4. Bill of materials

| Materials | Quantity | Cost (in Pesos) |
|---|---|---|
| Arduino Mega | 1 | 699 |
| PID DC motor Control: | | |
| DC motor w/ encoder | 1 | 645 |
| H-Bridge (TB6612FNG ) | 1 | 179 |
| PID DC motor positioning: | | |
| DC motor w/ encoder | 1 | 645 |
| Rotary Encoder | 1 | 38 |
| Ball and Beam Balancer: | | |
| Servo motor (MG996R) | 1 | 170 |
| Distance Sensor (Infrared) | 1 | 450 |
| PCB Fabrication (inc. shipping) | 5 pcs | 1380 |
| IP65 240mmx160mmx90mm enclosure | 1 | 350 |
| 3D Printing | 1 set | 1200 |
| | Total | **5756** |

As detailed in Table 4, the total cost of materials for the complete system is approximately ₱5,756, equivalent to around $103 USD (based on an estimated exchange rate of ₱56 = $1). The platform's core controller is an Arduino Mega priced at ₱699 (~$12.50), offering ample I/O capability at a low cost. The PID motor control setup includes a DC motor with an encoder (₱645 / ~$11.50) and an H-Bridge driver (TB6612FNG) (₱179 / ~$3.20). For the positioning module, a similar motor is paired with a rotary encoder (₱38 / ~$0.70). The ball-and-beam balancer uses an MG996R servo motor (₱170 / ~$3.00) and an infrared distance sensor (₱450 / ~$8.00). Fabrication and structural costs were also minimized through local PCB manufacturing (₱1,380 for five pieces / ~$24.60 total), an IP65-rated enclosure (₱350 / ~$6.25), and a complete 3D-printed frame (₱1,200 / ~$21.50), ensuring durability and modularity at low expense.

LabVIEW's graphics user interface (GUI) created an executable file for the project with a prerequisite LabVIEW RunTime 2018 Application installed in the user's machine. In the startup, four buttons are organized in the left portion, which play the roles of Home, The Team, Recommended Materials, and Feedback and Recommendation, respectively. The functions of each button are explained in seriatim. Refer to Fig. 5.

In-Home, this section redirects the user to the hardware functions described above. It also shows a message about whether the hardware is connected to the user's device, especially to the PIDs and Beam Balancer. Meanwhile, the calculator can operate without hardware communication. It consists of conversion between the Transfer Function (TF), State-Space Model (SSM), and Zeros-Poles-Gain Model (ZPKM) by plugging an input in any of the mentioned forms. It can also show the frequency and time response based on the given. Some indicators show the Phase and Gain Margin, Overshoot Time, and many more. Graphs of ZPKM, time sampling analysis, and root locus are also included ad hoc to grasp the standard numerical calculations in the feedback and control system. In addition, VI, which links to the hardware components, comprehensibly presents the graph, numerical indicator, and controls for the device of interest. Fig. 6 illustrates the VI Block Diagram in calling the PID and calculator VI files.

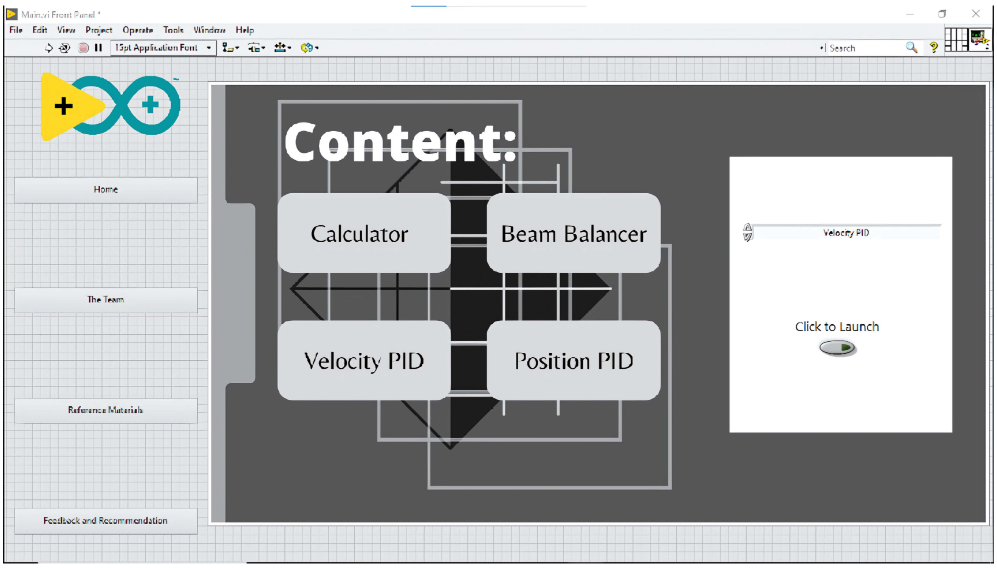


Fig. 5. LabVIEW main UI.

The remaining buttons relay to the mentioned functions. The contacts among the three are the feedback and recommendations, which link to the survey form that will be used to improve the project further.

## B. Simulated Results

This section presents the results obtained from the platform, covering: (1) the control system calculator, (2) PID simulations for position and velocity motors, and (3) the beam balancer.

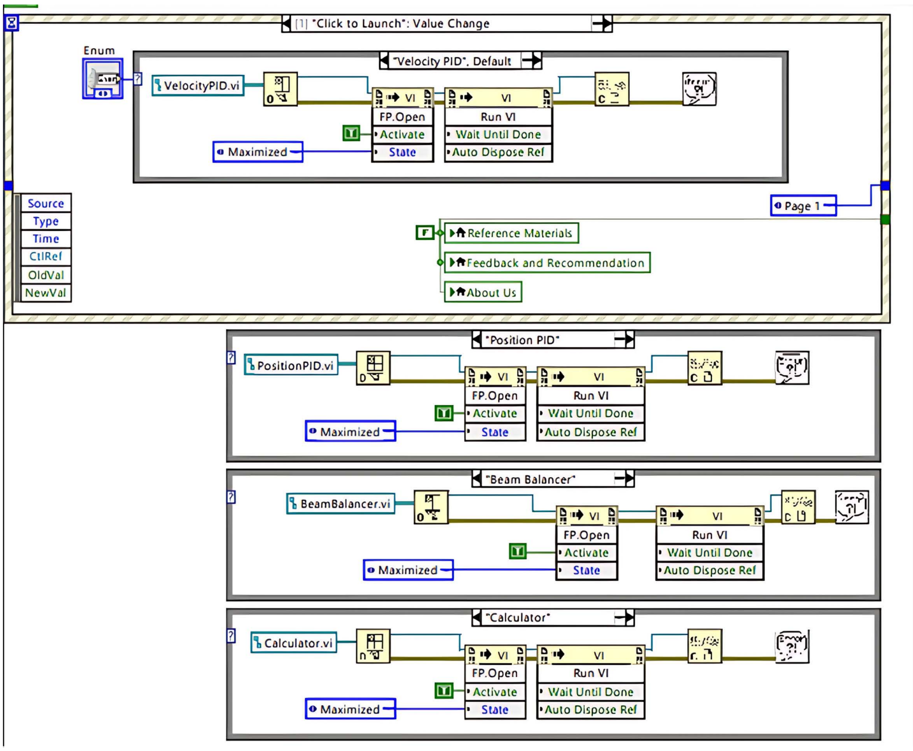

Fig. 6. LabVIEW main VI, responsible for redirecting to four functionalities, each split into distinct .vi files for enhanced modularity.

*1) Control system numerical computation example*

Consider, $F(s) = \frac{100}{s^2+15s+100}$, which is entered into the GUI via its numerator and denominator coefficients (see Fig. 7). The platform then calculates and displays the equivalent Zeros-Poles-Gain (ZPK) model and State-Space Model in the designated areas on the lower-left of the interface. The distinctness of the state-space model as transformed from the other two is observable because it is frequently presented in other forms, such as Canonical realization. There is also a slight difference in the results of some time response parameters, such as overshooting, which differs even in the tenth-place column, as shown in Fig. 7. Following a specific input characterization, like step or impulse input, any system output is said to change with respect to time. Generally, two phases would be involved in analyzing this response: the transient response, concerned with the immediate response, and the steady-state response, dealing with the system's long-term behavior. Some of the major response parameters are rise time, settling time, peak time, and overshoot, which give the user perspective on the system's speed, stability, and accuracy. The rest lie along textbook lines, including the frequency response in Fig. 8, which describes the response of the system to input signals of varying frequencies. Typically, this response is represented using Bode/Nyquist plots, and such representations are crucial in conditions of system stability and gain/phase margin. Such margin conditions can assist engineers in developing a design that will operate well across a range of operating conditions while not unstable conditions such as resonance or phase lag.

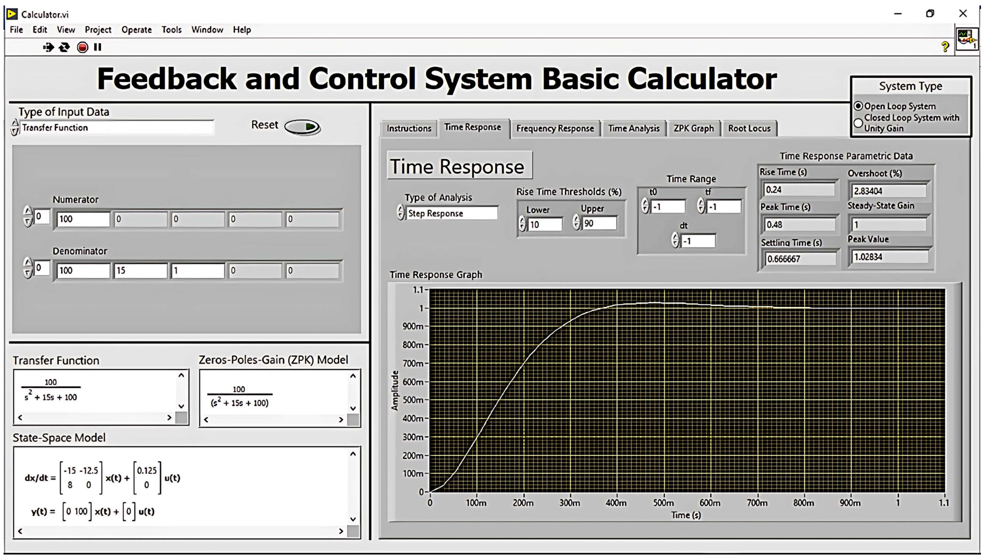

Fig. 7. The Time response value and graph of the sample transfer function alongside with equivalent transforms in the lower-left portion.

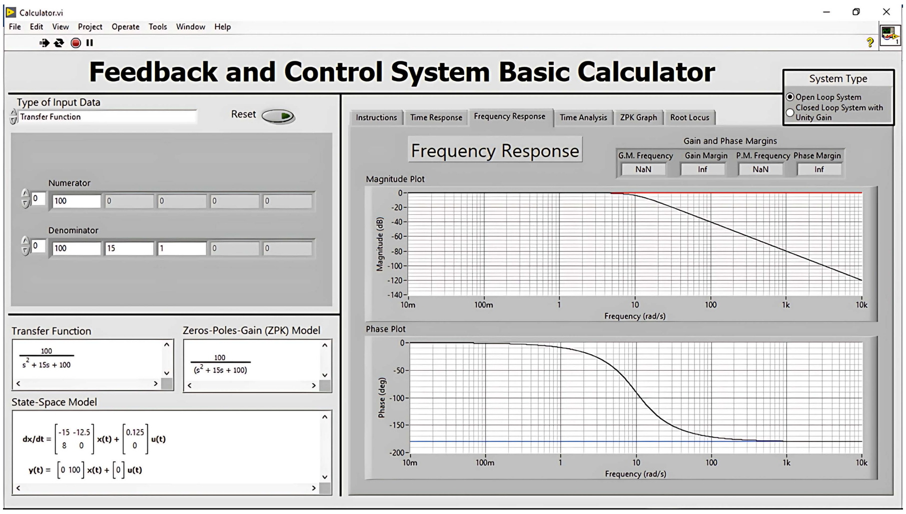


Fig. 8. The frequency response value and graph of the sample transfer function.

Then, we consider in Fig. 9 the zero-pole plot showing the graphical representation of zeros(s) such that the numerator is made 0 and poles(s) such that the denominator is made 0 of the system, projected to the complex plane. This is a critical visualization for predicting the system's stability, behavior, and responsiveness. A left-half plane pole signals stability, in contrast with poles on or near the imaginary axis that suggest oscillatory and unstable behavior. Finally, the time analysis in Fig. 10 shows a complete analysis of a system's behavior through time. Variable numerical inputs, along with plotted curves, are included to evaluate how output signals vary due to different inputs. With this methodology, engineers can compare actual system performance against its design criteria and make necessary adjustments to meet the desired specifications for real-world applications.

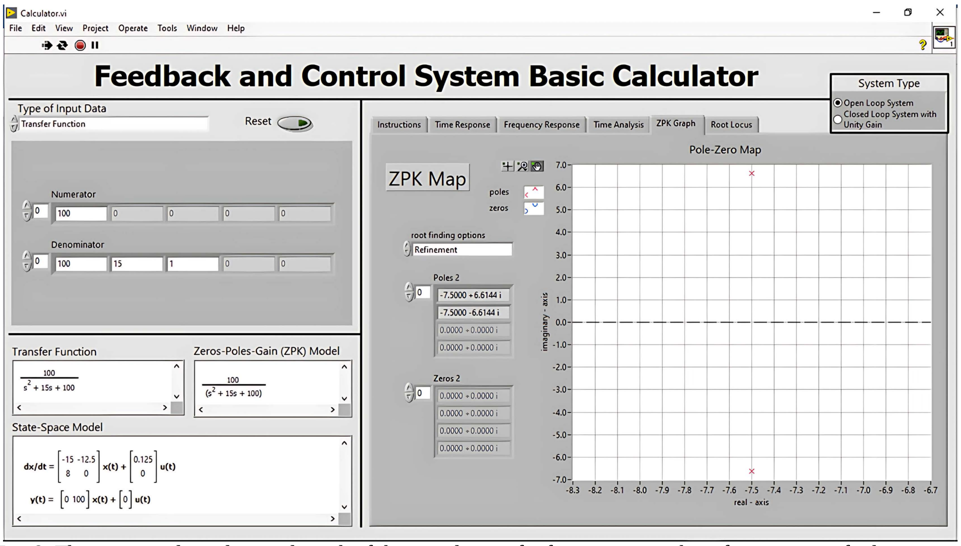


Fig. 9. The zeroes-poles values and graph of the sample transfer function using the refinement root finding option.

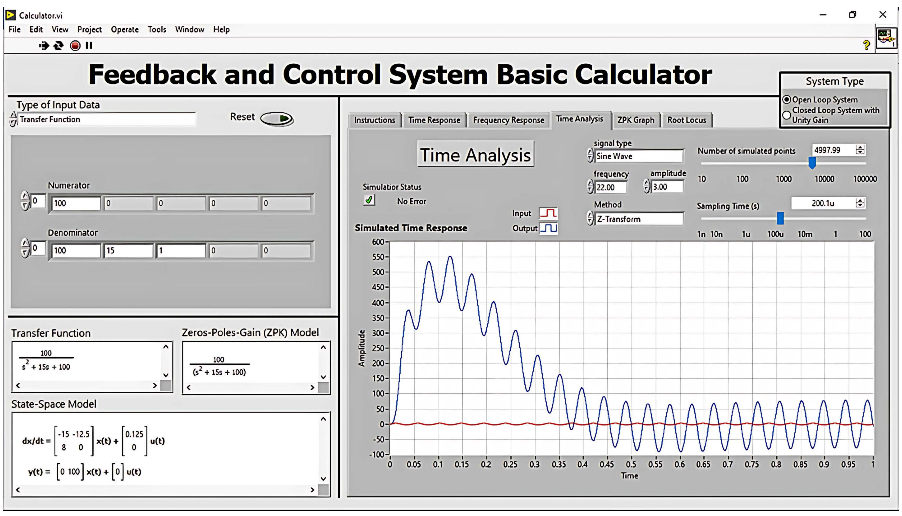


Fig. 10. Time analysis of the given transfer function at 22Hz sine wave signal type using the Z-transform method.

*2) PID simulations: Beam balancer, velocity, and position motors*

Fig. 11 provide the empirical data from the balancing beam control system. This system is designed for experimentation with real-time dynamic systems using a Proportional-Integral-Derivative controller.

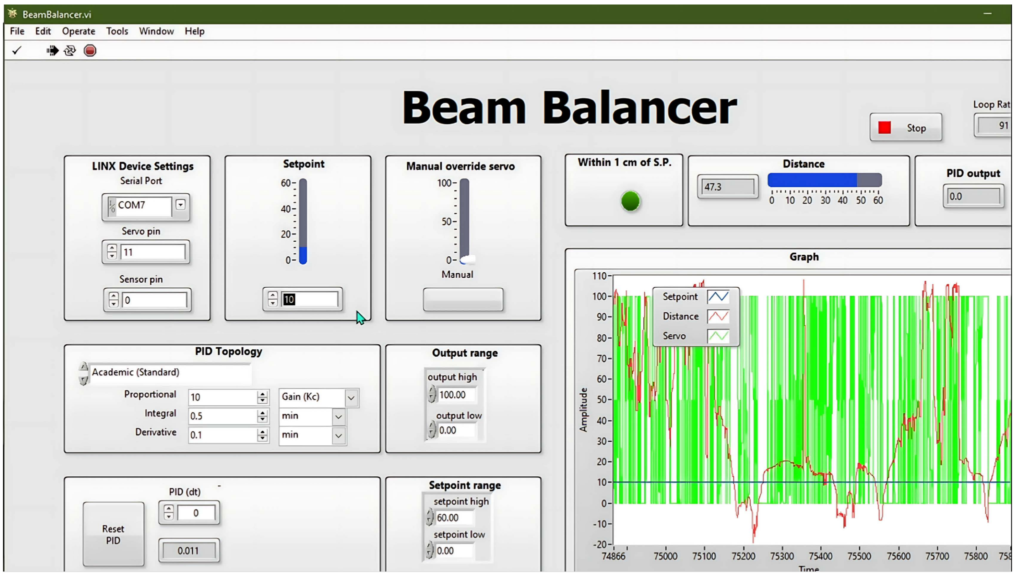


Fig. 11. Beam balancer software output at P=10, I = 0.5, and D = 0.1.

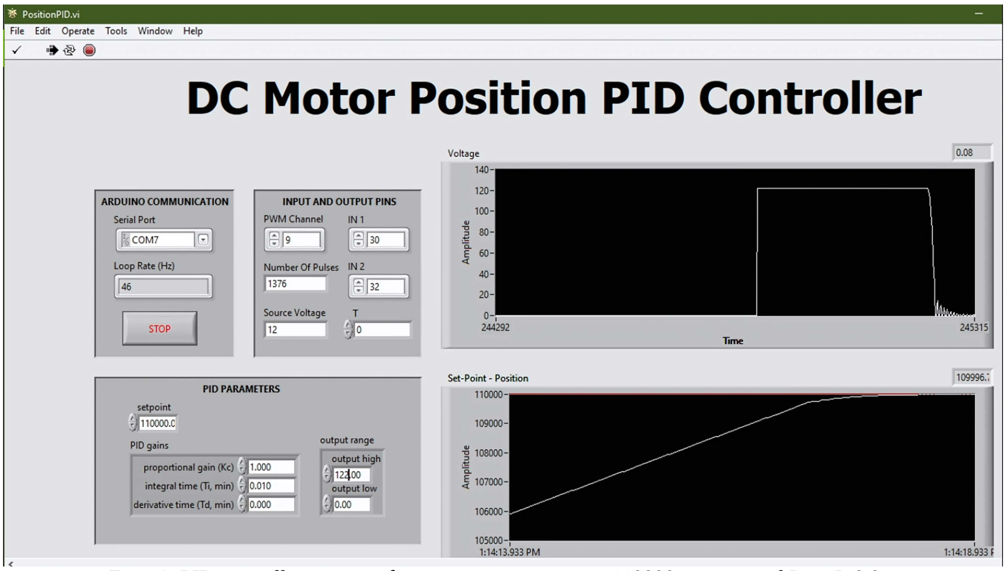


Fig. 12. PID controller outputs of motor position output at 110000 setpoint and P=1, I=0.01.

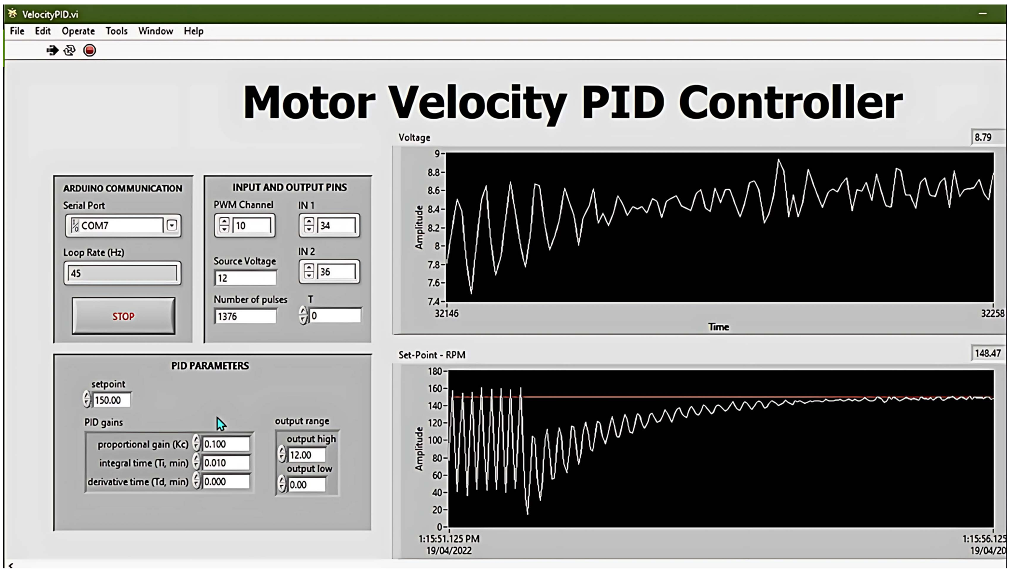


Fig. 13. PID controller output of Motor velocity output at 150 setpoint and P=0.1, I=0.01.

The platform includes several PID simulation modules. The beam balancer module (GUI shown in Fig. 11) utilizes a physical setup with a distance sensor and a servo motor interfaced with the Arduino. Separately, the platform provides simulation modules for a DC Motor Position PID Controller (GUI in Fig. 12) and a Motor Velocity PID Controller (GUI in Fig. 13), which also interface with the Arduino for controlling respective DC motors. The PID controller implemented in the LabVIEW environment controls the beam's position based on feedback from distance sensors and minimizes the distance of deviation from the reference. It adapts well to the equilibrium control of a beam balancer and serves as a flexible framework for studying control strategies that work on velocity. The system can also expand the application from static balance to dynamic positioning and motion control problems by coupling the PID control algorithm with the designated DC motors. The closed-loop response exhibits the strength of the PID controller in reacting to the inputs of position and velocity error signals with fast convergence, minimal overshoot, and steady-state error.

### *C. Acceptability Using PSSUQ*

The usability and acceptability of the developed LabVIEW-Arduino-based platform were evaluated using the Post-Study System Usability Questionnaire (PSSUQ). This tool, widely recognized for its reliability and precision, assesses key usability factors through four categories: system usefulness (SYSUSE), information quality (INFOQUAL), interface quality (INTERQUAL), and overall satisfaction (OVERALL). The survey responses from Electronics Engineering students provided insights into the platform's performance and user experience. Table 5 summarizes the findings, where lower mean scores signify higher usability and satisfaction.

Table 5. Result of PSSUQ

| PSSUQ Overall Category | Mean |
|---|---|
| System Usefulness (SYSUSE) | 2.30 |
| Information Quality (INFOQUAL) | 2.45 |
| Interface Quality (INTERQUAL) | 2.15 |
| Overall | 2.40 |

The results from the PSSUQ survey provide strong evidence supporting the efficacy of the developed LabVIEW-Arduino-based platform as an educational tool for control system engineering. System Usefulness (SYSUSE) achieved a mean score of 2.30, indicating that the platform offers a highly intuitive and user-friendly experience. Participants found the system straightforward to operate, which is critical for students unfamiliar with complex control systems or simulation environments. Tasks such as designing and analyzing PID controllers, interpreting zero-pole plots, and conducting state-space transformations were completed efficiently, demonstrating the platform's ability to simplify traditionally challenging concepts.

The Information Quality (INFOQUAL) score of 2.45 reflects the platform's effectiveness in delivering clear and accessible information. Features such as error message clarity, on-screen guidance, and supporting documentation were pivotal in enhancing the user experience. These tools reduced user frustration and empowered respondents to independently navigate and resolve issues during the experiments, fulfilling the educational objective of promoting self-directed learning.

The platform's Interface Quality (INTERQUAL) received a favorable rating (mean = 2.15), emphasizing the exceptional design of the graphical user interface (GUI). The GUI's structured layout, real-time visual indicators, and intuitive controls contributed significantly to the seamless execution of control simulations. This design approach aligns with best practices in usability engineering, ensuring that students remain engaged and focused on learning outcomes rather than system navigation. Overall Satisfaction (OVERALL) scored a mean of 2.40, affirming that the platform met or exceeded user expectations in delivering a robust and effective simulation environment. Respondents praised the integration of theoretical and practical aspects, noting how the platform bridges gaps between conceptual understanding and hands-on application cost-effectively.

## V. LIMITATIONS AND FUTURE DIRECTION

This study presents a promising low-cost platform that enhances control systems education using the LabVIEW-Arduino integration. However, several limitations must be acknowledged to provide a balanced understanding of the findings and outline improvement areas.

Firstly, the usability assessment was limited to 22 Electronics Engineering students from a single institution. While the feedback obtained was insightful and reflective of the participants' experiences, the small and homogeneous sample size may affect the generalizability of the results. Future research should aim to include a more diverse participant pool, across institutions and disciplines, to assess the platform's pedagogical effectiveness more comprehensively.

Secondly, the technical validation of the control system remains preliminary. The study demonstrates basic PID control applications but lacks rigorous performance analysis regarding control accuracy, response time, stability, and real-world applicability. To address this, future work will incorporate quantitative performance metrics such as Root Mean Square Error (RMSE), Integral Absolute Error (IAE), and settling time, as well as comparative performance benchmarking against established platforms like MATLAB/Simulink, Scilab/Xcos, and Python-control libraries.

The current implementation is limited to fundamental control algorithms, such as Proportional-Integral-Derivative (PID) control. Advanced strategies, including fuzzy logic, adaptive control, and model predictive control (MPC), were not explored. Integrating these approaches in future iterations would enhance the system's capability and expose students to a broader spectrum of modern control methods, better aligning the platform with real-world industry practices.

In terms of hardware, the platform is tied to specific configurations (e.g., Arduino Mega and selected sensors), which may limit scalability and adaptability. More complex control scenarios, such as nonlinear or multi-input multi-output (MIMO) systems, may exceed the processing capabilities of the current setup. Future work will explore alternative microcontrollers (e.g., Raspberry Pi, ESP32, or STM32) and modular designs that support more powerful computing and a wider variety of sensors and actuators to improve scalability and applicability. Additionally, expanding software compatibility beyond LabVIEW can increase integration flexibility in diverse learning environments. Given the current total cost of only ₱5,756 (~ $103), the system remains highly accessible; however, future comparisons will help refine its positioning among budget-conscious educational solutions.

From an educational perspective, while the Discussion section now explicitly links the platform's features and usage to pedagogical frameworks like constructivism and experiential learning, this initial study focused primarily on usability and technical feasibility. A key area for future work involves moving beyond usability assessments to rigorously evaluate the platform's impact on actual student learning outcomes, potentially using metrics aligned with Bloom's taxonomy or assessing problem-solving skills developed through PBL activities implemented using the platform. Further research could also explore integrating the platform within specific instructional models like flipped classrooms to optimize its pedagogical effectiveness and measure cognitive gains more formally.

In summary, while the proposed platform effectively meets its goal of delivering a low-cost and accessible tool for control systems education, these identified limitations serve as important directions for future enhancement, both technically and pedagogically, to ensure its long-term impact and sustainability in academic settings.

## VI. Discussion and Conclusion

This study successfully developed a low-cost, LabVIEW-Arduino integrated platform for control system computations and PID controller simulations, addressing critical educational needs in electronics engineering. The platform's ability to perform real-time numerical computations, including transfer function transformations and transient response analysis, and execute accurate PID simulations for beam balancers, velocity, and position control motors, underscores its practicality and educational value. By aligning with CHED CMO 101, series of 2017, the system ensures relevance to academic curricula while offering an affordable alternative to traditional laboratory setups.

This study developed and evaluated a low-cost, Arduino-LabVIEW-based educational platform to enhance student learning in control systems engineering. The platform's total development cost was approximately ₱5,756 (~$103 USD), demonstrating its significant cost-effectiveness compared to conventional laboratory setups. This affordability positions the platform as a viable instructional tool for institutions with limited laboratory resources, particularly in developing regions or resource-constrained academic settings.

The system's usability was assessed using the Post-Study System Usability Questionnaire (PSSUQ), with responses from 22 undergraduate Electronics Engineering students. The platform achieved an overall mean score of 2.40 on a 7-point Likert scale (lower scores indicating higher satisfaction). Subscale results were similarly favorable: 2.30 for system usefulness, 2.45 for information quality, and 2.15 for interface quality. These findings suggest a positive user experience, with students perceiving the system as intuitive, informative, and conducive to learning. Informal feedback further underscored the benefits of real-time interaction and hands-on experimentation in reinforcing theoretical concepts.

Beyond the positive usability metrics, the educational effectiveness of the LabVIEW-Arduino platform can be understood through its alignment with established pedagogical principles. The platform strongly supports Constructivist learning theory, where students actively build knowledge rather than passively receiving it. For instance, learners directly engage by manipulating PID controller gains (Kp, Ki, Kd) within the LabVIEW GUI and observing the immediate, real-time impact on system responses, visualized through graphs for the motor velocity/position simulations or reflected in the behavior of the physical beam balancer hardware. This iterative cycle of action, observation, and analysis allows students to construct a more profound, intuitive understanding of how controller parameters affect system stability and performance. Furthermore, the platform facilitates Experiential Learning by bridging the crucial gap between abstract theoretical concepts and tangible, practical applications. Students can apply principles learned in lectures, such as analyzing stability from pole locations using the platform's zero-pole plotting tool or predicting transient response characteristics (overshoot, settling time) and then validating these predictions through simulation or hardware interaction. This direct experience reinforces theoretical knowledge and enhances retention. The framework is also inherently suited for Problem-Based Learning (PBL) methodologies. Instructors can assign authentic engineering tasks, requiring students to 'design and tune a PID controller for the DC motor position control to achieve a settling time under 2 seconds with less than 10% overshoot,' using the platform as their testbed. Such activities promote critical thinking, design skills, and systematic problem-solving as students experiment with different control strategies to meet specified requirements. Engaging with the platform's diverse functionalities—from basic computations to simulation and hardware control —also encourages students to operate at multiple cognitive levels of Bloom's Taxonomy. They move from understanding control concepts via visualizations to applying tuning techniques, analyzing system performance based on graphical and numerical outputs, and potentially evaluating the trade-offs between different controller designs. While this study did not formally assess learning outcomes, the platform's design inherently supports these varied cognitive processes, underpinning its potential as an effective educational tool.

The results highlight that the system's user-friendly interface, real-time feedback, and seamless hardware-software integration provide an efficient and engaging educational tool. Nonetheless, the study acknowledges certain limitations, including the current implementation is confined to basic control strategies and lacks rigorous quantitative validation of system performance, including response time, accuracy, and robustness metrics. The fixed hardware configuration limits the system's scalability and applicability to more complex or industrial-grade control scenarios. Furthermore, the study did not benchmark the platform against alternative low-cost solutions, engage with broader educational theories, or address reproducibility considerations.

Future research should address these gaps by incorporating advanced control algorithms (e.g., fuzzy logic, model predictive control), expanding compatibility with diverse hardware platforms, and performing comparative performance analyses with other control system simulators. Aligning the system's instructional design with contemporary educational theories may also enhance its pedagogical impact. Additionally, making the platform open-source and widely accessible would promote reproducibility and collaborative development across institutions.

Finally, the proposed Arduino-LabVIEW platform offers a cost-effective, user-friendly, and educationally impactful approach to teaching control systems. With targeted refinements, it holds substantial promise as a scalable and inclusive solution for engineering education in both traditional and remote learning environments.

## DATA AVAILABILITY STATEMENT

The LabVIEW VIs, Arduino code, PCB design files (schematics and layout), 3D model files for the enclosure and beam balancer, and detailed bill of materials used in this study are available from the corresponding author upon reasonable request to support reproducibility and further research.

## CONFLICT OF INTEREST

The authors declare no conflict of interest

## AUTHOR CONTRIBUTIONS

R.G.G. contributed to the conceptualization, methodology, formal analysis, visualization, and writing of the original draft of the study. R.V.M. was responsible for investigation, and validation processes. Both R.G.G. and R.V.M. participated in writing—review and editing. All authors have read and approved the final manuscript.

## ACKNOWLEDGMENT

The researchers wish to convey their heartfelt gratitude to all individuals who generously assisted in completing this study. Special recognition is extended to Prince Jaminn Soberano, Allen Borja, John Benedict Nartates, and Krisha Cajigal for their valuable comments and recommendations, which significantly contributed to the improvement of this study. Additionally, we would like to express our profound appreciation for the unwavering support provided by Tarlac State University in successfully executing this research.

## REFERENCES

[1] F. Muratore, M. Gienger, and J. Peters, "Assessing Transferability from Simulation to Reality for Reinforcement Learning," *IEEE Trans Pattern Anal Mach Intell*, vol. 43, no. 4, pp. 1172–1183, Jul. 2019. doi: 10.1109/TPAMI.2019.2952353

[2] J. Drgoňa *et al.*, "All you need to know about model predictive control for buildings," *Annu Rev Control*, vol. 50, pp. 190–232, Jan. 2020. doi: 10.1016/J.ARCONTROL.2020.09.001

[3] J. Collins, D. Howard, and J. Leitner, "Quantifying the reality gap in robotic manipulation tasks," in *Proc IEEE Int Conf Robot Autom*, vol. 2019-May, pp. 6706–6712, May 2019. doi: 10.1109/ICRA.2019.8793591

[4] I. Uyanik and B. Catalbas, "A low-cost feedback control systems laboratory setup via Arduino–Simulink interface," *Computer Applications in Engineering Education*, vol. 26, no. 3, pp. 718–726, May 2018. doi: 10.1002/CAE.21917

[5] IEEE Robotics and Automation Society and Institute of Electrical and Electronics Engineers, presented at 2017 IEEE International Conference on Mechatronics and Automation, IEEE ICMA 2017, August 6–9, 2017, Takamatsu, Japan.

[6] I. Ishafit, T. K. Indratno, and Y. D. Prabowo, "Arduino and LabVIEW-based remote data acquisition system for magnetic field of coils experiments," *Phys Educ*, vol. 55, no. 2, 2020. doi: 10.1088/1361-6552/ab5ed6

[7] F. J. Jiménez-Romero, J. R. González-Jiménez, F. García-Torres, Á. Caballero, and F. R. Lara-Raya, "A novel testing equipment based on Arduino and LabVIEW for electrochemical performance studies on experimental cells: Evaluation in lithium-sulfur technology," *Measurement (Lond)*, vol. 224, 2024. doi: 10.1016/j.measurement.2023.113922

[8] V. León-Martínez, E. Peñalvo-López, J. Montañana-Romeu, C. Andrada-Monrós, and L. Molina-Cañamero, "Assessment of load losses caused by harmonic currents in distribution transformers using the transformer loss calculator software," *Environments—MDPI*, vol. 10, no. 10, 2023. doi: 10.3390/environments10100177

[9] T. M. Guzmán-Armenteros, J. Ruales, J. Villacís-Chiriboga, and L. S. Guerra, "Experimental prototype of electromagnetic emissions for biotechnological research: Monitoring cocoa bean fermentation parameters," *Foods*, vol. 12, no. 13, 2023. doi: 10.3390/foods12132539

[10] N. A. Othman, N. S. Damanhuri, M. A. Syafiq Mazalan, S. A. Shamsuddin, M. H. Abbas, and B. C. Chiew Meng, "Automated water quality monitoring system development via LabVIEW for aquaculture industry (Tilapia) in Malaysia," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 20, no. 2, pp. 805–812, Nov. 2020. doi: 10.11591/ijeecs.v20.i2.pp805-812

[11] A. D. Rincon-Quintero, L. A. D. Portillo-Valdes, and C. L. Sandoval-Rodriguez, "Evaluation of a cocoa dryer prototype using LabVIEW software and Arduino hardware," *Heritage and Sustainable Development*, vol. 5, no. 2, pp. 363–374, 2023. doi: 10.37868/hsd.v5i2.318

[12] A. Kavga, V. Thomopoulos, E. Pischinas, D. Tsipianitis, and P. Nikolakopoulos, "Design and simulation of a greenhouse in a computational environment (ANSYS/FLUENT) and an automatic control system in a LABVIEW environment," *Simul Model Pract Theory*, vol. 129, 2023. doi: 10.1016/j.simpat.2023.102837

[13] H. Dillon, N. Schmedake, K. E. Eifler, T. A. Doughty, and K. Lulay, "Design of a curriculum-spanning mechanical engineering laboratory experiment," in *Proc. ASEE Annual Conference and Exposition, Conference Proceedings*, 2016.

[14] S. Haryoko and H. Jaya, "Development of LabVIEW in microcontroller subjects at Vocational School Industrial Electronics Engineering Department," *Journal of Engineering and Applied Sciences*, vol. 13, no. 11, pp. 4150–4155, 2018. doi: 10.3923/jeasci.2018.4150.4155

[15] H. M. Asraf, K. A. N. Dalila, M. Y. Zakiah, Z. A. Amar Faiz, and M. T. Nooritawati, "Computer assisted e-laboratory using LabVIEW and internet-of-things platform as teaching aids in the industrial instrumentation course," *International Journal of Online Engineering*, vol. 14, no. 12, pp. 26–42, 2018. doi: 10.3991/ijoe.v14i12.8992

[16] A. Cavagnino, G. Pellegrino, A. Estebsari, E. Armando, and R. Bojoi, "A new laboratory for hands-on teaching of electrical engineering," *IEEE International Symposium on Industrial Electronics*, 2018, pp. 883–889. doi: 10.1109/ISIE.2018.8433848

[17] P. Ponce, A. Molina, O. Mata, and G. Baltazar, "LEGO® EV3 platform for STEM education in elementary school," in *ACM International Conference Proceeding Series*, 2019, pp. 177–184. doi: 10.1145/3318396.3318426

[18] A. Sahoo and Y. Chang, "Laboratory activities of the fundamentals of mechatronics course for undergraduate engineering technology students," in *Proc. ASEE Annual Conference and Exposition*, 2019.

[19] J. Aguilera-Alvarez, J. Padilla-Medina, C. Martínez-Nolasco, V. Samano-Ortega, M. Bravo-Sanchez, and J. Martínez-Nolasco, "Development of a didactic educational tool for learning fuzzy control systems," *Math Probl Eng*, vol. 2021, 2021 doi: 10.1155/2021/3158342

[20] H. Fukumoto, T. Yamaguchi, M. Ishibashi, and T. Furukawa, "Developing a remote laboratory system of stepper motor for learning support," *IEEE Transactions on Education*, vol. 64, no. 3, pp. 292–298, 2021, doi: 10.1109/TE.2020.3042595

[21] P. Y. Qaraei, A. Hashemi, M. H. Salehi, M. Karimi, and A. Moradi, "DC machines remote lab case study in technical and vocational university," *Computer Applications in Engineering Education*, 2022. doi: 10.1002/cae.22530

[22] R. El Dahr, X. Lignos, S. Papavieros, and I. Vayas, "Development and validation of a LabVIEW automated software system for displacement and dynamic modal parameters analysis purposes," *Modelling*, vol. 4, no. 2, pp. 189–210, 2023. doi: 10.3390/modelling4020011

[23] I. P. Idehen and Q. You, "Development of a low-cost seismograph with trigger system for geoscience education," *Instruments and*

*Experimental Techniques*, vol. 66, no. 2, pp. 358–363, 2023. doi: 10.1134/S0020441223020173

[24] P. K. Kumar Reddy, A. Rashmi, B. V. Adarsha, and M. Govinda Raju, "Low-cost, robust data acquisition system for automotive testing and validation using LabVIEW," in *Proc. IEEE Region 10 Annual International Conference*, 2023, pp. 612–616. doi: 10.1109/TENCON58879.2023.10322392

[25] M. S. Rahman, M. Saha, M. M. Hossain, A. K. M. Nazrul Islam, and H. Monir, "Design of a real time remote monitoring and controlling system using LabVIEW," in *Proc. 2023 14th International Conference on Computing Communication and Networking Technologies, ICCCNT 2023*, 2023. doi: 10.1109/ICCCNT56998.2023.10306933

[26] M. A. Kuhail, S. Farooq, R. Hammad, and M. Bahja, "Characterizing visual programming approaches for end-user developers: A systematic review," *IEEE Access*, vol. 9, pp. 14181–14202, 2021. doi: 10.1109/ACCESS.2021.3051043

[27] Commission on Higher Education, *BS_ECE_CMO-101-s.-2017-BS-Electronics-Engineering*, 2017.

[28] J. Kodosky and C. Lopes, "LabVIEW," in *Proc. the ACM on Programming Languages*, vol. 4, no. HOPL, Jun. 2020. doi: 10.1145/3386328

[29] S. M. Kim, Y. Choi, and J. Suh, "Applications of the open-source hardware Arduino platform in the mining industry: A review," *Applied Sciences 2020*, vol. 10, no. 14, p. 5018, Jul. 2020. doi: 10.3390/APP10145018

[30] A. Polo, P. Narvaez, and C. Robles Algarín, "Implementation of a cost-effective didactic prototype for the acquisition of biomedical signals," *Electronics 2018*, vol. 7, no. 5, p. 77, May 2018. doi: 10.3390/ELECTRONICS7050077

[31] N. Montes, N. Rosillo, M. C. Mora, and L. Hilario, "A novel real-time MATLAB/Simulink/LEGO EV3 platform for academic use in robotics and computer science," *Sensors 2021*, vol. 21, no. 3, p. 1006, Feb. 2021. doi: 10.3390/S21031006

[32] M. Saad, A. H. Amhedb, and M. Al Sharqawi, "Real time DC motor position control using PID controller in LabVIEW," *Journal of Robotics and Control (JRC)*, vol. 2, no. 5, pp. 342–348, Sep. 2021. doi: 10.18196/JRC.25104

[33] B. Dudem *et al.*, "Wearable triboelectric nanogenerator from waste materials for autonomous information transmission via morse code," *ACS Appl Mater Interfaces*, vol. 14, no. 4, pp. 5328–5337, Feb. 2022, doi: 10.1021/ACSAMI.1C20984/ASSET/IMAGES/LARGE/AM1C20984_0005.JPEG.

[34] F. J. Jimenez, F. R. Lara, and M. D. Redel, "API for communication between Labview and Arduino UNO," *IEEE Latin America Transactions*, vol. 12, no. 6, pp. 971–976, Sep. 2014. doi: 10.1109/TLA.2014.6893988

[35] T. Q. Duc, "LabVIEW and open embedded system," *LabVIEW—A Flexible Environment for Modeling and Daily Laboratory Use*, Jul. 2021. doi: 10.5772/INTECHOPEN.98271

[36] H. Muhammad Asraf, K. A. N. Dalila, A. W. M. Hakim, and R. H. M. F. Hon, "Development of experimental simulator via Arduino-based PID temperature control system using LabVIEW," *Journal of Telecommunication, Electronic and Computer Engineering (JTEC)*, vol. 9, no. 1–5, pp. 53–57, Apr. 2017.

[37] D. Calinoiu, R. Ionel, M. Lascu, and A. Cioabla, "Arduino and LabVIEW in educational remote monitoring applications," in *Proc. Frontiers in Education Conference, FIE*, Feb. 2015. doi: 10.1109/FIE.2014.7044027

[38] E. J. Alfuth, T. Chester, N. Roisen, C. Valtierrez, V. Bagonza, and B. L. Stottrup, "A diy langmuir trough made with Adruino, LabVIEW, and 3D printed parts for education and research," *Biophys J*, vol. 108, no. 2, p. 333a, Jan. 2015. doi: 10.1016/J.BPJ.2014.11.1816

[39] F. A. M. Allawi, "Educational interactive virtual strain and consolidation tests using LabVIEW and Arduino Uno microcontroller," *Computer Applications in Engineering Education*, vol. 31, no. 5, pp. 1256–1271, 2023. doi: 10.1002/cae.22638

[40] B. Jakubiec and K. Sokoliński, "Educational manipulation robot," *AUTOBUSY—Technika, Eksploatacja, Systemy Transportowe*, vol. 20, no. 1–2, pp. 336–339, Feb. 2019. doi: 10.24136/ATEST.2019.062

[41] P. Y. Qaraei, A. Hashemi, M. H. Salehi, M. Karimi, and A. Moradi, "DC machines remote lab case study in Technical and Vocational University," *Computer Applications in Engineering Education*, vol. 30, no. 5, pp. 1440–1453, Sep. 2022. doi: 10.1002/CAE.22530

[42] P. Aradi, "Offline and online thermostat experiment with LabVIEW and Arduino," in *Proc. 2016 International Symposium on Small-Scale Intelligent Manufacturing Systems, SIMS 2016*, pp. 127–131, Dec. 2016. doi: 10.1109/SIMS.2016.7802912

[43] Z. Peng, L. Ma, and F. Xia, "A low-cost embedded controller for complex control systems," in *Proc. the 5th International Conference on Embedded and Ubiquitous Computing, EUC 2008*, 2008, pp. 23–29. doi: 10.1109/EUC.2008.40

[44] M. Kalúz, Ľ. Čirka, R. Valo, and M. Fikar, *ArPi Lab: A Low-cost Remote Laboratory for Control Education*.

[45] W. Ferry and A. Otieno, *Development of a Low Cost Laboratory System for Teaching Automation System Integration in the Manufacturing Engineering Technology Curriculum*, 2004.

[46] C. N. Galvis, H. C. Medina, and J. Á. Ariza, "Design and implementation of a low-cost remote tank control system for engineering and technology education," in *Proc. 7th International Symposium on Accreditation of Engineering and Computing Education, ICACIT 2021*, Institute of Electrical and Electronics Engineers Inc., 2021. doi: 10.1109/ICACIT53544.2021.9612512

[47] C. Yfoulis and S. Papadopoulou, "A portable low-cost Arduino-based laboratory kit for control education," in *Proc. 2018 UKACC 12th International Conference on Control (CONTROL)*, Sheffield, United Kingdom, Sep. 2018, p. 435.

[48] I. Uyanik and B. Catalbas, "A low-cost feedback control systems laboratory setup via Arduino–Simulink interface," *Computer Applications in Engineering Education*, vol. 26, no. 3, pp. 718–726, May 2018. doi: 10.1002/cae.21917

[49] C. C. Peng, "Microprocessor-based virtual control system framework for control engineering education," in *Proc. 2024 IEEE 7th Eurasian Conference on Educational Innovation: Educational Innovations and Emerging Technologies, ECEI 2024*, Institute of Electrical and Electronics Engineers Inc., 2024, pp. 414–419. doi: 10.1109/ECEI60433.2024.10510786

[50] T. Klinger, C. Kreiter, A. Pester, and C. Madritsch, "Low-cost remote laboratory concept based on NI myDAQ and NI Elvis for electronics engineering education," in *Proc. 2020 IEEE Global Engineering Education Conference*, Porto, Portugal, Apr. 2020, pp. 106–109.

[51] J. R. Lewis, "Psychometric evaluation of the post-study system usability questionnaire: The PSSUQ," in *Proc. the Human Factors Society Annual Meeting*, vol. 36, no. 16, pp. 1259–1260, Oct. 1992. doi: 10.1177/154193129203601617

[52] R. A. Malik and M. R. Frimadani, "Lean UX: Applied PSSUQ to evaluate less-ON UI/UX analysis and design," *International Journal of Advances in Data and Information Systems*, vol. 4, no. 1, pp. 73–85, Apr. 2023. doi: 10.25008/ijadis.v4i1.1263

[53] K. Moulaei, R. Moulaei, and K. Bahaadinbeigy, "The most used questionnaires for evaluating the usability of robots and smart wearables: A scoping review," *Digit Health*, vol. 10, 20552076241237384, Jan. 2024. doi: 10.1177/20552076241237384

[54] Y. Natrio, R. A. Malik, and R. Kasrin, "Uncovering user experience evaluation in nursing-English several platforms through SUS and PSSUQ," vol. 9, no. 4, pp. 2403–2417, 2024. doi: 10.29100/jipi.v9i4.6780